

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Denis Rajković

**Spletna stran za razstavo
novomedijske umetnosti**

DIPLOMSKO DELO

INTERDISCIPLINARNI UNIVERZITETNI
ŠTUDIJSKI PROGRAM PRVE STOPNJE
RAČUNALNIŠTVO IN MATEMATIKA

MENTOR: izr. prof. dr. Narvika Bovcon

Ljubljana, 2017

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Razstavi umetniških del v galerijskem prostoru s spletno stranjo povečamo časovni in prostorski doseg. Novomedijska umetnost je zaradi zastarevanja tehnologije, ki jo poganja, posebej občutljiva in težko jo je hraniti. Spletna stran s fotografskim in video gradivom, izvornimi kodami projektov ter opisnimi besedili je eden od načinov arhiviranja te vrste umetnosti. V diplomski nalogi izdelajte dvojezično spletno stran za razstavo Sreča Dragana Prostor, vržen iz tira ter jo postavite na laboratorijski strežnik. Razstava v Muzeju sodobne umetnosti Metelkova je predstavila okoli 60 projektov, ki so nastajali v petih desetletjih avtorjevega ustvarjanja. Projekti so bili tematsko in kronološko razporejeni po sobah.

Zahvaljujem se mentorici izr. prof. dr. Narviki Boucon za njen čas, potrpežljivost, hitro odzivnost ter vsa mnenja in napotke pri izdelavi diplomske naloge. Pri tem se za vso pomoč zahvaljujem tudi asistentu Žigi Emeršiču. Preostalo zahvalo namenjam družini, prijateljem in sošolcem, ki so me ves čas podpirali in mi stali ob strani.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Tehnologije	3
2.1	HTML5	3
2.2	CSS	4
2.3	JavaScript in jQuery	4
2.4	EJS	5
2.5	Bootstrap	5
2.6	MongoDB in Mongoose	6
2.7	Node.js in Express	7
3	Priprava	9
3.1	Ogled razstave	9
3.2	Primeri dobre prakse	10
3.3	Gradivo	19
4	Izdelava spletne strani	21
4.1	Odziven dizajn	22
4.2	Minimalizem	22
4.3	Znak	23
4.4	Navigacija	28

4.5	Gradnja podatkovne baze	29
4.6	Interaktivni tloris	31
4.7	Pogledi	36
5	Optimizacija in izboljšave	43
5.1	Slike	43
5.2	Videi	45
6	Vzdrževanje in predlogi za prihodnost	53
7	Zaključek	55
	Literatura	57

Seznam uporabljenih kratic

kratica	angleško	slovensko
HTML	Hyper Text Markup Language	hipertekstovni označevalni jezik
CSS	Cascading Style Sheets	kaskadne stilske podloge
WWW	World Wide Web	svetovni splet
API	Application Programming Interface	aplikacijski programski vmesnik
DOM	Document Object Model	objektni model dokumenta
JSON	JavaScript Object Notation	JavaScript objektna notacija
NoSQL	Not Only Structured Query Language	ne samo strukturirani povpraševalni jezik
DBMS	Database Management System	sistem za upravljanje podatkovnih baz
ODM	Object Data Model	objektni podatkovni model
OS	Operating System	operacijski sistem
HTTP	Hypertext Transfer Protocol	protokol za prenos hiperteksta
URL	Uniform Resource Locator	enolični krajevnik vira

MVC	Model View Controller	način zasnove arhitekture aplikacije
SVG	Scalable Vector Graphics	skalabilna vektorska grafika
PDF	Portable Document Format	prenosni format dokumenta
JPEG	Joint Photographic Experts Group	združena skupina fotografskih ekspertov
PNG	Portable Network Graphics	prenosne omrežne grafike
MPG	Moving Picture Experts Group	Skupina ekspertov za premikajoče se slike
TS	Transport Stream	transportni tok
AVI	Audio Video Interleave	avdio video preplet
TCP	Transmission Control Protocol	protokol za nadzor prenosa
IP	Internet Protocol	internetni protokol
ACC	Advanced Audio Coding	napredno avdio kodiranje
NPM	Node Package Manager	Node upravitelj paketov

Povzetek

Naslov: Spletna stran za razstavo novomedijske umetnosti

Avtor: Denis Rajković

Cilj diplomskega dela je bila izdelava spletne strani za razstavo Sreča Dragana, Prostor, vržen iz tira, ki je bila na ogled v Muzeju sodobne umetnosti Metelkova od 22. 12. 2016 do 5. 3. 2017. Razstava je zajemala petdesetletno zgodovino Draganove medijske umetnosti in njen razvoj v slovenskem kulturnem prostoru. V diplomu opišemo, kako smo zbrali in uredili gradivo za spletno stran ter kako smo medije najrazličnejših oblik prevedli v formate, ki se jih da ogledati na spletu. Razložimo tudi postopek oblikovanja, kako smo strani optimizirali in kako smo dosegli, da stran ustrezno in kar se da celostno dokumentira razstavo ter jo nadgrajuje s spletno izkušnjo.

Ključne besede: spletna stran, novomedijska umetnost, video, oblikovanje, arhiviranje medijske umetnosti, odzivne spletne strani.

Abstract

Title: Website for new media art exhibition

Author: Denis Rajković

The goal of the diploma was to create a website for the exhibition of Srečo Dragan Space is out of joint, which was on display at the Museum of Contemporary Art Metelkova from 22 December 2016 to 5 March 2017. The exhibition covered the 50-year history of Dragan's media art and its development in the Slovenian cultural space. In the diploma, we describe how we collected and edited the material for the website, and how we translated the media of various forms into formats that can be viewed on the web. We also explain the process of designing, how we optimized the website and how we achieved that the website is appropriately and as a whole documenting the exhibition and upgrading it with a web experience.

Keywords: website, new media art, video, design, archiving media art, responsive websites.

Poglavje 1

Uvod

V diplomskem delu smo izdelali spletno stran za razstavo Sreča Dragana *Prostor, vržen iz tira*, ki je bila na ogled v Muzeju sodobne umetnosti Metelkova od 22. 12. 2016 do 5. 3. 2017. Razstava je zajemala petdesetletno zgodovino Draganove medijske umetnosti in njen razvoj v slovenskem kulturnem prostoru.

„Dela iz šestdesetih in sedemdesetih let, ko je umetnik deloval bodisi sam bodisi v sodelovanju z Nušo Dragan, so pomenila pomemben prispevek takratni neoavantgardni umetnosti in pozneje k t. i. „novi umetniški praksi“. Kot avtorja razstav, programov in tekstov sta lastnemu delu in videu poskušala zagotoviti ustrezno kontekstualizacijo znotraj širšega jugoslovanskega in mednarodnega prostora. Začetki avtorjevega delovanja na področju videa so povezani s prakso udejanjanja umetnosti kot ideje skupine OHO. Še zlasti so pomembna pionirska dela na področju interaktivne in medijske umetnosti, ko je televizijski program bil pomemben medij produkcije, reprezentacije in refleksije avtorskega videa. Dragan je v osemdesetih in devetdesetih letih nadaljeval z raziskovanjem možnosti, ki jih ponujajo novi mediji, video ter mrežna in kibernetična tehnologija. Za njegova iskanja iz devetdesetih let je pomembna obiskovalčeva izkušnja, ki temelji na tehnološko podprtem razširjenem videnju. Svojo novejšo produkcijo usmerja v realizacije interaktivnih instalacij in računalniške 3D animacije [33].“

S spletno stranjo smo uporabnikom hoteli zagotoviti čim bolj podobno izkušnjo, kot bi jo dobili, če bi si razstavo ogledali v živo. Želeli smo doseči, da bodo Draganova umetniška dela uporabnikom na razpolago, kadarkoli si jih bodo želeli ogledati, kot tudi da jih predstavimo širši publiki. V začetku diplome opisujemo vse stvari, ki jih je bilo potrebno narediti pred pričetkom izdelave spletne strani. Začnemo z ogledom razstave, sledi analiza dobrih primerov spletnih strani s podobno tematiko, kot je bila naša, nato pa razložimo še, kako smo zbrali in uredili gradivo, ki ga prikažemo na strani. Naslednje poglavje govori o izdelavi spletne strani. Tukaj opišemo pristop odzivnega dizajna, lastnosti minimalističnega sloga in zakaj je pomembno, da ima stran tudi svoj znak. Spustimo se v nekatere tehnične stvari, razložimo funkcionalnosti pomembnejših komponent in opišemo posamezne poglede spletne strani. Temu sledi poglavje o optimizaciji in izboljšavah, za konec pa razložimo še, kako se spletno stran vzdržuje in kakšni so naši predlogi za prihodnost.

Poglavje 2

Tehnologije

2.1 HTML5

„HTML je standardni označevalni jezik za izdelavo spletnih strani.“ HTML dokumenti se spremenijo v spletne strani, ko jih spletni brskalnik prejme od strežnika. „HTML opiše strukturo spletne strani semantično [8].“ Koda HTMLja je sestavljena iz HTML elementov (npr. `<p></p>`). HTML elementi so znaki, ki se nahajajo med špičastimi oklepaji. „Običajno so sestavljeni iz dveh značk: uvodne značke (npr. `<p>`) in zaključne značke (npr. `</p>`).“ Zaključna značka vsebuje poleg špičastih oklepajev tudi poševnico. Vsak HTML element pove brskalniku informacijo, ki je shranjena med uvodno in zaključno značko. V uvodni znački lahko najdemo tudi tako imenovane attribute (npr. `<p lang="en-us">`), ki nam podajo dodatno informacijo o vsebini elementov. „Sestavljeni so iz dveh delov: imena (npr. `lang`) in vrednosti (npr. `en-us`), ločenih z enačajem.“ Ime atributa nam pove, katero dodatno informacijo nosi o vsebini elementa in je napisan z malo začetnico. Vrednost atributa predstavlja informacijo oz. nastavitev atributa in je napisana v narekovajih. „Večino atributov se lahko uporablja le na določenih elementih, vendar jih obstaja tudi nekaj, ki jih lahko uporabljamo na vsakemu elementu [39].“ HTML5 je peta verzija in trenutno najnovejša verzija HTML standarda. „Izdana je bila z namenom izboljšanja podpore najnovejše

multimedije ter sočasno omogoča berljivost s strani človeka in računalniških naprav. S slednjo verzijo HTMLja so prišli na plan tudi novi elementi, s katerimi lažje upravljamo grafično vsebino [10].“ Eden izmed njih je element `<video>`, v katerega smo se poglobili v izdelavi diplomske naloge.

2.2 CSS

„CSS so kaskadne stilske podloge. Uporabljamo jih za prezentacijo dokumentov, napisanih v označevalnem jeziku [3].“ CSS nam omogoča, da ustvarimo pravila, ki določajo, kako naj vizualno izgleda vsebina HTML elementov (npr. `p {color: red;}`). Posamezno pravilo je sestavljeno iz dveh delov: selektorja (npr. `p`) in deklaracije (npr. `color: red`). „Selektor nam pove, na katere elemente se navezuje pravilo. Isto pravilo lahko velja tudi za druge elemente, če so imena elementov navedena in ločena z vejico (npr. `h1, h2, h3, p`). Deklaracija nam pove, kako naj bodo oblikovani elementi, navedeni v selektorju. CSS deklaracija se nahaja v zavutih oklepajih in je sestavljena iz dveh delov: lastnosti (npr. `color`) in vrednosti (npr. `red`). V eni deklaraciji je možno navesti več lastnosti, ki jih ločimo s podpičjem (npr. `p {color: red; font-family: Arial;}`). Lastnost določa, kateri vidik elementa želimo spremeniti, vrednost pa nam pove, katero nastavitve želimo izbrati za izbrano lastnost [39].“ CSS3 je trenutno najnovejša verzija CSSja, ki poskuša razširiti funkcionalnost prejšnje verzije CSS 2.1. Vsebuje kar nekaj dolgo pričakovanih novosti, kot so „zaobljeni robovi, sence, gradienti in mnogo drugih [4].“ V nadaljevanju diplome smo izkoristili večino slednjih novosti s poudarkom na animaciji.

2.3 JavaScript in jQuery

JavaScript je visoko nivojski, dinamični, objektno orientiran programerski jezik. Poleg HTMLja in CSSja, je JavaScript eden izmed treh glavnih tehnologij svetovnega spleta. Med drugim se ga uporablja za izdelavo interaktivnih

spletnih strani. Jezik je podprt s strani vseh modernih spletnih brskalnikov, brez potrebe po dodatnih vtičnikih zaradi že vgrajenega JavaScript motorja. „Ima API za delo s teksti, tabelami, datumi, regularnimi izrazi in osnovno manipulacijo DOMa.“ V začetku se je JavaScript uporabljal samo na strani odjemalca v spletnih brskalnikih, dan danes pa ga najdemo tudi na „strani strežnika, podatkovnih bazah, kot tudi v nespletnih programih [13].“

„jQuery je JavaScript knjižnica, ustvarjena z namenom, da poenostavi programiranje HTMLja na strani odjemalca.“ Je brezplačna, odprtokodna in največkrat uporabljena JavaScript knjižnica na spletu. Njena sintaksa je narajena tako, da je z njo „lažje upravljati dokumente, izbrane DOM elemente, ustvarjati animacije, obvladovati dogodke in izvajati AJAX klice.“ JQuery omogoča tudi, da „razvijalci ustvarijo svoje vtičnike na podlagi JavaScript knjižnice.“ Rezultat slednjega je, da jQuery omogoča gradnjo močnih in dinamičnih spletnih strani [15].

2.4 EJS

EJS je jezik na strani uporabnika, s katerim lahko kreiramo predloge. „Originalno je bil EJS del JavaScriptMVCja, ki ga je sedaj nadomestil DoneJs.“ EJS omogoča kombinacijo podatkov in predlog, iz česar dinamično ustvari HTML strani. Če želimo v predlogi izvajati JavaScript kodo, jo napišemo med znaka `<% in %>`, če želimo njeno vrednost, pa med znaka `<%= in %>`. „Sintaksa tega jezika je nastala na osnovi ERubya in je podobna PHPju.“ Jezik EJS smo izbrali, ker omogoča hitro parcialno gradnjo strani s preprostim pošiljanjem spremenljivk pogledom [5] [14].

2.5 Bootstrap

„Bootstrap je brezplačno, odprtokodno spletno ogrodje za oblikovanje spletnih strani. Vsebuje predloge za HTML elemente, CSS pravila, tipografijo, forme, gumbe, navigacijo ter ostale komponente za kakovosten uporabniški

vmesnik.“ „Vsebuje tudi podaljšek JavaScripta v obliki jQuery vtičnika“, le ta pa vsebuje nekaj dodatnih elementov (npr. okno za dialog) za uporabniški vmesnik ter doda nekaj funkcionalnosti k že obstoječim (npr. avtomatsko dopolnjevanje vnesenih besed). Ena izmed ključnih lastnosti Bootstrapa je, da omogoča odziven spletni dizajn, zato je Bootstrap eno izmed najbolj popularnih ogrodij na svetu. Odziven spletni dizajn pomeni, da se postavitev spletne strani dinamično prilagaja velikosti zaslona, na katerem je prikazana. Na naši spletni strani smo uporabili verzijo Bootstrap 3. Navedena verzija je bila prva, ki je preoblikovala komponente v stilu ploskovnega dizajna in uporabila mobile-first (sl. prednost imajo mobilne naprave) pristop [1]. V času pisanja diplome je bila že prosto dostopna novejša verzija Bootstrap 4, ki je bila še v beta verziji in je zato nismo uporabili v našem projektu.

2.6 MongoDB in Mongoose

MongoDB je odprtokodni, NoSQL sistem za upravljanje podatkovnih baz. „Podatke shranjuje v dokumentih oblike JSON, ki se lahko razlikujejo v strukturi. Sorodne informacije so shranjene skupaj za hitre poizvedbe z MongoDB poizvedbenim jezikom. MongoDB uporablja dinamične sheme, kar pomeni, da dopušča ustvarjanje dokumentov, ne da bi vnaprej definirali strukturo, polja ali tipe njihovih vrednosti. Dovoljuje tudi spreminjanje strukture dokumenta s preprostim dodajanjem novih polj ali brisanjem že obstoječih. Podatkovni model omogoča predstavitev hierarhičnih povezav za preprosto shranjevanje tabel in drugih kompleksnejših struktur. Dokumenti v kolekciji morajo imeti identičen nabor polj in pogosto prihaja do denormalizacije podatkov. MongoDB je bil ustvarjen z namenom široke dostopnosti in skalabilnosti [20].“

„Mongoose je ODM knjižnica, ki omogoča dosledno modelarsko okolje za podatke. Vsiljuje potrebno strukturo ter sočasno omogoča veliko fleksibilnost MongoDBja [30].“

2.7 Node.js in Express

„Node.js je platforma na strani strežnika, zgrajena na Google Chromovemu JavaScript motorju (V8 motor). Je odprtokodno okolje za razvijanje aplikacij na strani strežnika in omrežnih aplikacij. Node.js aplikacije so napisane v JavaScript jeziku in lahko delujejo v času delovanja Node.jsa na OS X, Microsoft Windows in Linux. Node.js omogoča tudi bogato knjižnico številnih JavaScript modulov, ki poenostavijo razvoj spletnih aplikacij.“ Vsi APIji Node.jsa so asinhroni, kar pomeni, da strežnik, ki uporablja Node.js nikoli ne čaka APIja, da mu ta vrne podatke, ampak se pomakne na naslednji API., Node.js uporablja eno nitni model s kroženjem dogodkov.“ Aplikacije, ki uporabljajo Node.js, nikoli ne shranjujejo podatkov v medpomnilnik, ampak jih izpisujejo v kosih [28].

„Express je minimalno in fleksibilno Node.js ogrodje za spletne aplikacije, ki omogoča robusten nabor funkcionalnosti za razvoj spletnih in mobilnih aplikacij.“ Olajšuje tudi hiter razvoj spletnih aplikacij na osnovi Nodea. Express.js omogoča „nastavitev vmesnega sistema za odgovore na HTTP pozivedbe“. „Definira tabelo omrežnih poti, ki se uporablja za različne akcije glede na HTTP metodo in URL. Omogoča tudi dinamično gradnjo HTML strani glede na poslane argumente v predloge [7].“

Poglavje 3

Priprava

3.1 Ogled razstave

Preden smo se lotili načrtovanja spletne strani, smo si najprej ogledali razstavo. Razstava je bila postavljena v Muzeju sodobne umetnosti Metelkova v Ljubljani. Naslov razstave je bil „Prostor, vržen iz tira“. Na njej smo si lahko ogledali približno šestdeset del Sreča Dragana, ki so nastajala v petdesetih letih njegove medijske ustvarjalnosti. Celotna razstava je bila razdeljena v 8 sob. Na vhodu razstave nas je pričakala velika črna stena s kratkim opisom razstave v slovenščini in angleščini. Na steni smo lahko videli tudi mrežo, ki je prikazovala povezave med projekti, ki smo jih lahko videli na razstavi. Obiskovalec je imel na tej točki dve možnosti, če je vstopil na razstavo skozi desni vhod je začel ogled z Draganovimi prvimi deli, v kolikor je izbral levi vhod pa je lahko najprej videl njegove najnovejše projekte. V sobah so bili projekti razporejeni tako, da je bilo okrog posameznega projekta dovolj prostora in si ga je lahko naenkrat ogledovalo več ljudi, ne da bi motili drug drugega. Videli smo paleto različnih tipov projektov, od konceptualnih tabel, videov, do interaktivnih instalacij. Z ogledom razstave smo lahko dobili občutek, kako obsežna bo morala biti spletna stran in v kakšnem slogu jo bo potrebno oblikovati.

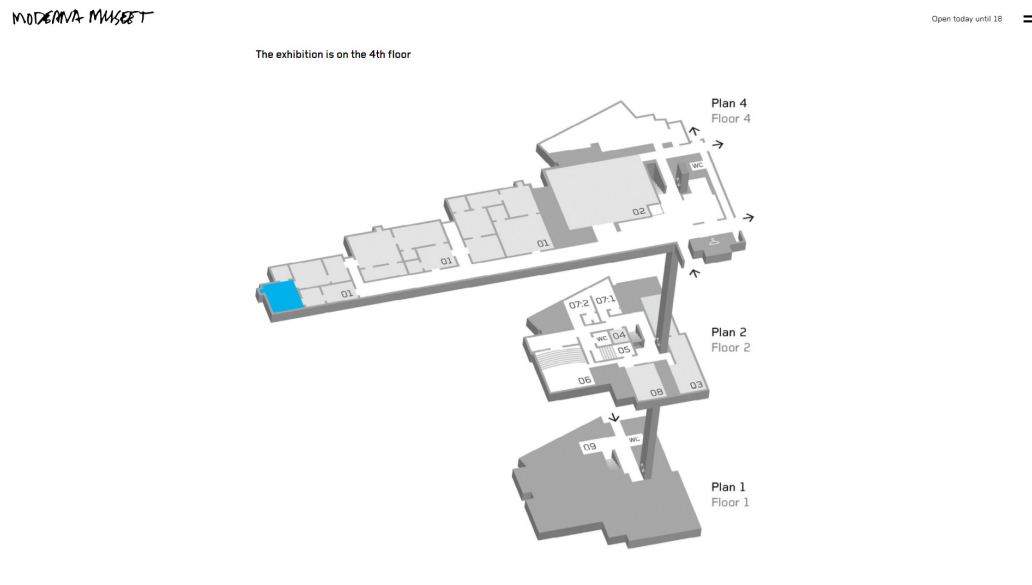
3.2 Primeri dobre prakse

Za izdelavo dobre spletne strani je potrebno nekaj časa vložiti tudi v raziskavo in analizo dobrih primerov na spletu. S tem procesom lahko že v začetku dela ugotovimo, kaj deluje dobro na spletnih straneh – s tematiko, podobno naši – in čemu se je dobro izogniti. Pri tem ne ocenjujemo le lepote in vizualnega izgleda spletne strani, temveč tudi njeno uporabnost, ter kakšno uporabniško izkušnjo omogoča obiskovalcu. Da bi dobili dober pregled nad različnimi oblikovalskimi rešitvami smo si na spletu ogledali strani svetovno znanih umetnikov, muzejev, galerij in razstav. Na vsaki strani smo izpostavili in opisali dobre implementacije.

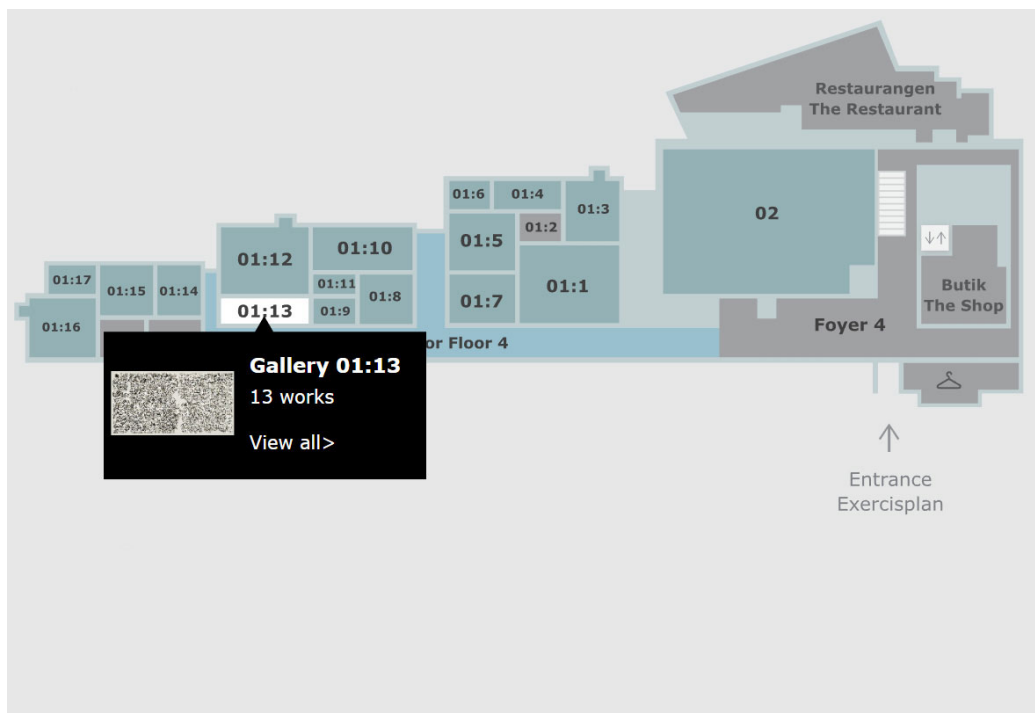
3.2.1 Moderna Museet

„Moderna Museet je vladna organizacija z nacionalno odgovornostjo za zbiranje, ohranjanje in razstavljanje moderne in sodobne umetnosti. Sedež ima v Stockholmu in Malmö [18].“ „Hrani dela Pabla Picassa, Ljubova Popova, Salvadorja Dalíja ter mnogih drugih znanih umetnikov. Skupaj hrani kar 6000 slik, skulptur in inštalacij, 25000 akvarelov, risb in tiskov, 400 videov in filmov ter 100000 fotografij [19].“ Spletna stran je dostopna na naslovu <http://www.modernamuseet.se/stockholm/en/> [17]. Stran vsebuje kar nekaj modernih funkcionalnosti, ki jih tipično ne vidimo na spletnih straneh. Na podstraneh za posamezno razstavo ima obiskovalec možnost, da mu opis razstave bere kar sama spletna stran. Možno si je tudi nastaviti barve za že prebrani tekst, ter prilagoditi hitrost branja. Poleg tega ima stran za vsako razstavo tudi sliko, v katerem nadstropju in prostoru muzeja se nahaja (slika 3.1), kar je zagotovo zelo uporabno za navigacijo v tako velikih muzejih. Na enem izmed pogledov je možno najti tudi interaktivni načrt muzeja, ki omogoča uporabniku, da si s klikom na posamezno galerijo ogleda vsa dela z opisi, ki so razstavljeni v njej (slika 3.2). Spletna stran se prilagaja različno velikim zaslonom, zato nudi dobro uporabniško izkušnjo ne glede na to, s katero napravo dostopamo do nje. Narejena je v minimalističnem slogu,

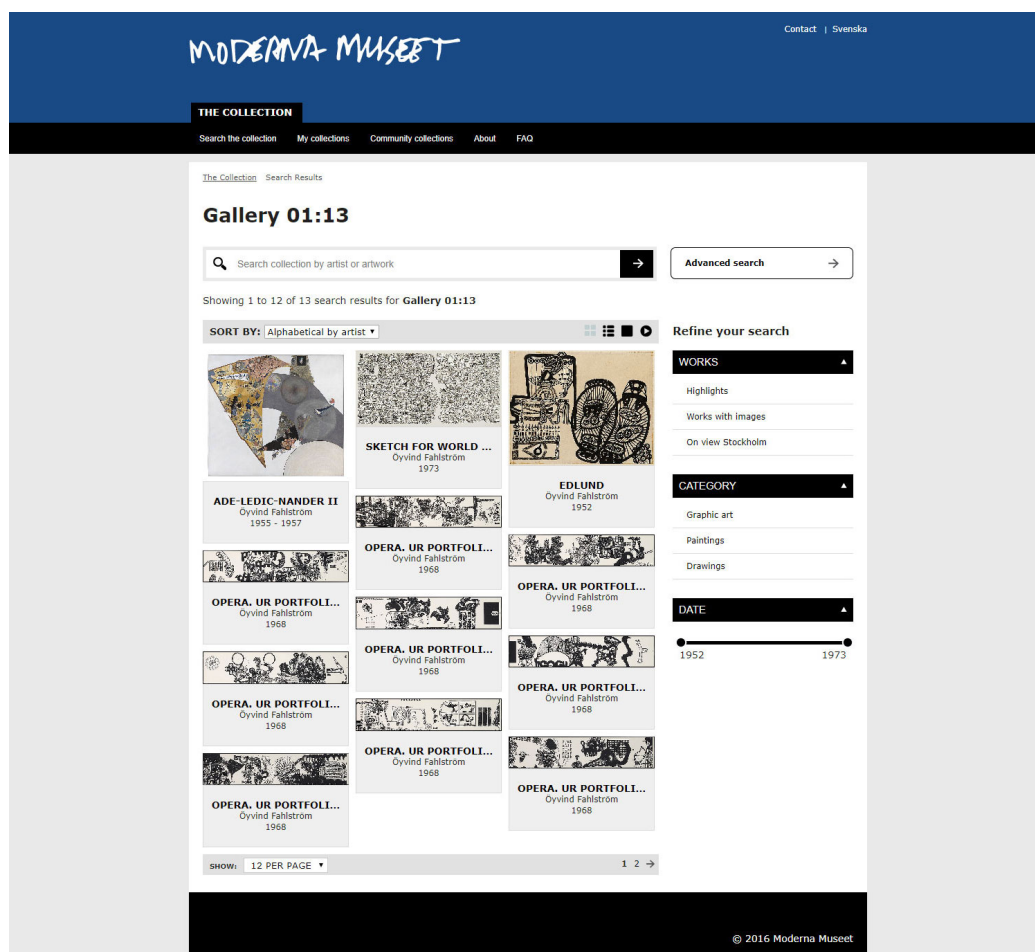
vsebina je postavljena v sredini spletne strani z veliko praznega prostora ob straneh.



Slika 3.1: Prikaz lokacije razstave v muzeju. Spletna stran organizacije Moderna Museet.



Slika 3.2: Interaktivni tloris muzeja. Spletna stran organizacije Moderna Museet.



Slika 3.3: Pregled del v izbrani galeriji. Spletna stran organizacije Moderna Museet.

3.2.2 Tate

„Tate je inštitut, ki shranjuje nacionalne kolekcije Britanske moderne in sodobne umetnosti. Je mreža štirih muzejev umetnosti: Tate Britain, Tate Liverpool, Tate St Ives Cornwall in Tate Modern [35].“ Spletna stran inštituta je dostopna na naslovu <http://www.tate.org.uk/> [34]. „Že od nastanka spletne strani leta 1988 stran zagotavlja informacije o vseh štirih muzejih pod isto domeno.“ Na strani lahko najdemo tudi veliko gradiva za izobraževanje, kot so številne „ure arhiviranih spletnih prenosov različnih dogodkov, članki iz časopisov,“ razne e-learning priložnosti in mnogo drugega [35]. Stran omogoča tudi, da si obiskovalec lahko načrtuje ogled razstave pred odprtjem. Na podstrani posameznega projekta lahko najdemo sekcijo s podrobnim opisom in ostalimi pomembnimi informacijami o projektu. Najdemo lahko ime avtorja, dimenzije umetnine, leto nakupa, ime kolekcije ter ostale podrobne informacije (slika 3.4). Kar je vredno izpostaviti, je tudi sekcija na dnu pogleda, ki uporabniku ponudi ogled projektov, za katere sklepa, da bi mu bili lahko všeč (slika 3.5). Tudi ta stran je narejena v minimalističnem slogu z veliko praznine ob robovih. Ponovno lahko tudi vidimo postavitev strani na belem ozadju s teksti temnejših barv.

Display caption	Catalogue entry		
<p>This was painted in the summer of 1911 as part of a decorative scheme for the dining room at the Borough Polytechnic, at the Elephant and Castle, London. The theme of the room's decoration was 'London on Holiday', and Grant responded by painting this idealised panorama of seven male nudes bathing. The nudes also represented the continuous movement of a single figure. Their bodies, like the water, are stylised to heighten the decorative effect of the image. Grant's depiction of the male body was greatly influenced by Michelangelo's ink studies and in particular his cartoon 'Battle of Cascina'. The celebration of the male nude was underpinned by both artists by a personal homoerotic fascination.</p> <p><i>Gallery label, February 2010</i></p>		ON DISPLAY AT TATE BRITAIN Exhibition: <i>Queer British Art 1861-1967</i>	
		ARTIST	Duncan Grant 1885–1978
		MEDIUM	Oil paint on canvas
		DIMENSIONS	Support: 2286 x 3061 mm
		COLLECTION	Tate
		ACQUISITION	Purchased 1931
		REFERENCE	N04567

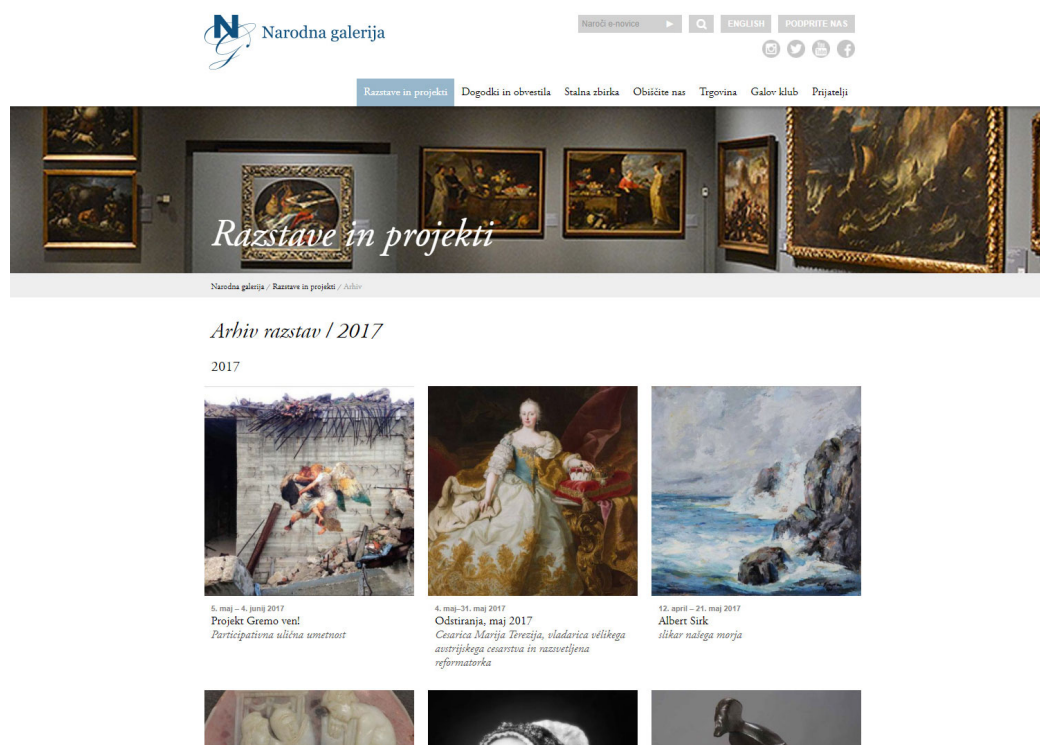
Slika 3.4: Opis projekta. Spletna stran inštituta Tate.



Slika 3.5: Sekcija s ponujenimi projekti. Spletna stran inštituta Tate.

3.2.3 Narodna galerija Slovenije

„Narodna galerija je osrednja nacionalna ustanova za starejšo umetnost v Sloveniji, ki hrani največjo zbirko likovnih del na slovenskem ozemlju od visokega srednjega veka do 20. stoletja [29].“ Spletna stran galerije je dostopna na naslovu <http://www.ng-slo.si/si/> [25]. Na podstrani razstav in projektov vidimo nekaj elementov, ki jih še nismo izpostavili na prejšnjih primerih. Prva stvar, ki jo lahko omenimo, je komponenta, ki ji pravimo drobtinice. Uporabniku kaže, kje se trenutno nahaja v hierarhiji spletne strani in preko katerih podstrani je prišel do sem. S to komponento se uporabnik lažje orientira na strani, kar zagotovo pripomore k boljši uporabniški izkušnji. K temu pripomore tudi gumb za hiter dvig na vrh spletne strani, ki se uporabniku prikaže v desnemu spodnjemu kotu, ko se začne pomikati navzdol po strani. Poleg tega lahko opazimo tudi zelo lep primer urejene razporeditve vsebine. Projekti so razporejeni v tri stolpce, pri čemer je vsak projekt predstavljen s kratkim opisom in majhno predstavitevno sliko. Vsak projekt je predstavljen le s ključnimi informacijami, kar je tudi eno izmed načel minimalističnega sloga (slika 3.6). Poleg tega ponovno kot v prejšnjih primerih opazimo belo podlago, tekst črne barve in veliko praznega prostora.



Slika 3.6: Spletna stran Narodne galerije Slovenija.

3.3 Gradivo

Preden smo začeli z dejansko izdelavo spletne strani, smo morali zbrati in urediti še vso gradivo, ki ga bomo prikazali na strani. Izkazalo se je, da je bil to eden izmed dejavnikov priprave spletne strani, za katerega smo porabili največ časa. Večino gradiva smo prejeli od mentorice in avtorja razstave. Gradivo je zajemalo tekstovne datoteke s podatki o projektih, fotografije postavitve razstave in projektov, filme, videe ter razne slike v vektorski obliki. Nekaj manjkajočih videov smo morali tudi poiskati in prenesti iz aplikacije Mouseion Serapeion. Gre za aplikacijo, ki hrani nekatere projekte avtorjev, ki so sodelovali s Fakulteto za računalništvo in informatiko. Ko smo imeli vse gradivo zbrano na enem mestu, ga je bilo potrebno še urediti.

Odločili smo se, da projekte uredimo po sobah tako, kot so bili razstavljeni na razstavi. Začeli smo s tem, da smo za vsako sobo narisali skico tlorisa in na območje, kjer se je nahajal posamezni projekt, napisali ID kot številko. Prva cifra je predstavljala številko sobe, preostale pa številko projekta. Nato je bilo potrebno urediti podatke iz tekstovnih datotek. Opazili smo, da se podatki o projektih zelo razlikujejo glede na tip projekta. Zaradi tega smo si najprej izpisali vse različne informacije, ki jih lahko najdemo o posameznemu projektu. Našli smo sledeče: avtorje projekta, ime projekta, letnico izdelave, medij, stopnjo rekonceptualizacije, dolžino medija (v kolikor je bil projekt video, smo imeli podatek o časovni dolžini), prizorišče, ime lastnika in opis. Potem smo v Excelovem programu naredili tabelo, kjer je vsaka vrstica predstavljala podatke posameznega projekta, vsak stolpec pa je vseboval eno izmed informacij, ki smo jih izpisali v prejšnjem koraku. O vsakem projektu nismo imeli vseh možnih informacij, ker so bili projekti različnih tipov in zaradi tega, ker so bili podatki pomanjkljivi. Zaradi slednjega je bila končna tabela zelo luknjičasta. Na prvo mesto tabele smo dodali stolpec, v katerega smo vnesli še IDje projektov. Ker smo za posamezni projekt imeli več različnih fotografij in videov, smo se odločili, da za vsak projekt ustvarimo datotečno mapo z IDjem projekta in v njo prenesli ustrezne fotografije ali videe. S tem smo dosegli, da smo lahko preko IDja

vedeli, v kateri sobi se je nahajal projekt, preko tabele, katere informacije o projektu so na voljo, in preko datotečnih map, na katerih fotografijah in videih smo ga lahko videli. Slikovno in video gradivo smo pri izdelavi spletne strani večkrat preoblikovali, mu spreminjali format, velikosti in podobno, zato dokončno pripravo tega gradiva opišemo v zadnjem delu diplome, kjer opisujemo tudi, kako smo ga optimizirali.

Poglavje 4

Izdelava spletne strani

Na tej točki smo si že ogledali razstavo, naredili temeljito analizo dobrih spletnih strani in zbrali vse gradivo, ki smo ga želeli predstaviti na strani. Sledila je dejanska izdelava spletne strani. Odločili smo se, da uporabimo MVC vzorec. MVC je arhitekturni vzorec programske opreme. Sprva se je pojavil „pri razvoju uporabniških vmesnikov za namizne računalnike, danes pa ga uporabljamo tako pri razvoju spletnih aplikacij“ kot tudi spletnih strani. Celotna ideja MVCja je, da razdeli arhitekturo na 3 dele: model, pogled in kontroler. Model lahko opišemo kot „metodo za dostopanje in spreminjanje podatkov in stanja, pogled prikazuje vsebino modela uporabniku v ustreznem vmesniku, kontroler pa služi kot posrednik med modelom in pogledom.“ Takšna razdelitev porazdeli kompleksnost izdelave med omenjene dele ter s tem doseže, da je „vsak del zase manj kompleksen.“ MVC vzorec tudi „poveča fleksibilnost, saj je vsak del spletne strani lažje prilagoditi kakršnimkoli spremembam“ in ga lahko razvijamo ločeno od ostalih delov [24]. Ker smo posamezne dele tekom izdelave večkrat spremenili, v nadaljevanju opisujemo le končno rešitev naše spletne strani.

4.1 Odziven dizajn

V današnjem času se tehnologija razvija zelo hitro. Vsakodnevno prihajajo na tržišče novi računalniki, tablice in telefoni zato je zelo pomembno, da je spletna stran narejena tako, da je prijazna do uporabnika ne glede na to, s katere naprave dostopa do nje. In prav to omogoča odziven dizajn, saj poskrbi, da ima uporabnik nadzor nad spletno stranjo ne da bi moral pri tem uporabljati dodatne aktivnosti kot npr. zoomiranje, obračanje ekrana, pomikanje levo, desno in podobno, saj se spletna stran sama prilagaja glede na to, s katere naprave dostopamo do nje in s tem naredi uporabniško izkušnjo bolj prijetno [32]. Kot smo že omenili, smo za našo spletno stran uporabili ogrodje Bootstrap. Ker je ogrodje že narejeno tako, da se prilagaja različno velikim zaslonom, potrebujemo zelo malo dela, da dosežemo odzivnost na vseh pogledih. Bootstrap je eno izmed najbolj popularnih ogrodji za oblikovanje spletnih strani, zato lahko na spletu najdemo zelo veliko dokumentacije za implementacijo različnih funkcionalnosti, ki jih omogoča. K hitrejšemu razvoju pripomore tudi to, da z ogrođjem pridejo že narejeni gradniki, ki nam zelo olajšajo začetek gradnje spletne strani in postavitev temeljnih elementov.

4.2 Minimalizem

Glede na analizo spletnih strani s podobno tematiko, kot je naša, smo prišli do zaključka, da je zelo popularen minimalističen dizajn. Minimalizem kot umetniški stil se je razvil v času po drugi svetovni vojni in se še danes pojavlja v modernih spletnih dizajnih. Zagovarja pojem „manj je več“, kar se odraža v izgledu spletne strani z veliko „negativnega prostora, neokrašeno tipografijo in malo elementi.“ Največji izziv pri uporabi tega sloga je, kako narediti spletno stran preprosto in hkrati zanimivo. Najpomembnejše štiri karakteristike minimalizma so: reducirana vsebina, negativni prostor, vizualna harmonija (simetrija) in kontrast [6]. Pri izdelavi spletne strani smo se neprestano sklicevali na slednje štiri karakteristike in sočasno pazili, da stran

ni izgledala preveč dolgočasno. Da smo dosegli reducirano vsebino, smo se vedno vprašali, kaj je bistvo, ki ga želimo prikazati na trenutnem pogledu. V kolikor element in njegova vsebina nista prispevala k temu cilju, smo ga odstranili. Tako smo na strani dobili manj elementov, pri čemer je vsak prišel bolj do izraza in posledično tudi njegova vsebina. Implementacija negativnega prostora je bila dosežena z uporabo Bootstrapovega CSS razreda container. Slednji drži elemente v centru strani z negativnim prostorom ob straneh. Centriranje vsebine in ne izkoriščanje celotnega zaslona daje vtis kompaktnosti in pušča prostor eleganci. Simetrija je izmed vseh karakteristik edina, ki je nismo uspeli izpopolniti na vseh pogledih. Do težav je prihajalo, kadar smo imeli opravka z lihim številom elementov in se jih ni dalo razporediti v sodo število stolpcev. Pri večini primerov iz analize spletnih strani smo opazili, da je ozadje strani bele barve, zato smo tudi naše ozadje naredili v tej barvi. Izbor bele barve je vedno dobra izbira, kadar želimo doseči močan kontrast med barvami vsebine in barvo ozadja. S tem hkrati dosežemo tudi, da je bolj razvidno, kje se posamezna komponenta spletne strani začne in kje konča ter posledično, da celotna vsebina pride bolj do izraza. Kontrast smo poudarili tudi s tem, da smo vse obrobe, črte in pisave obarvali s črno ali temnim odtenkom sive.

4.3 Znak

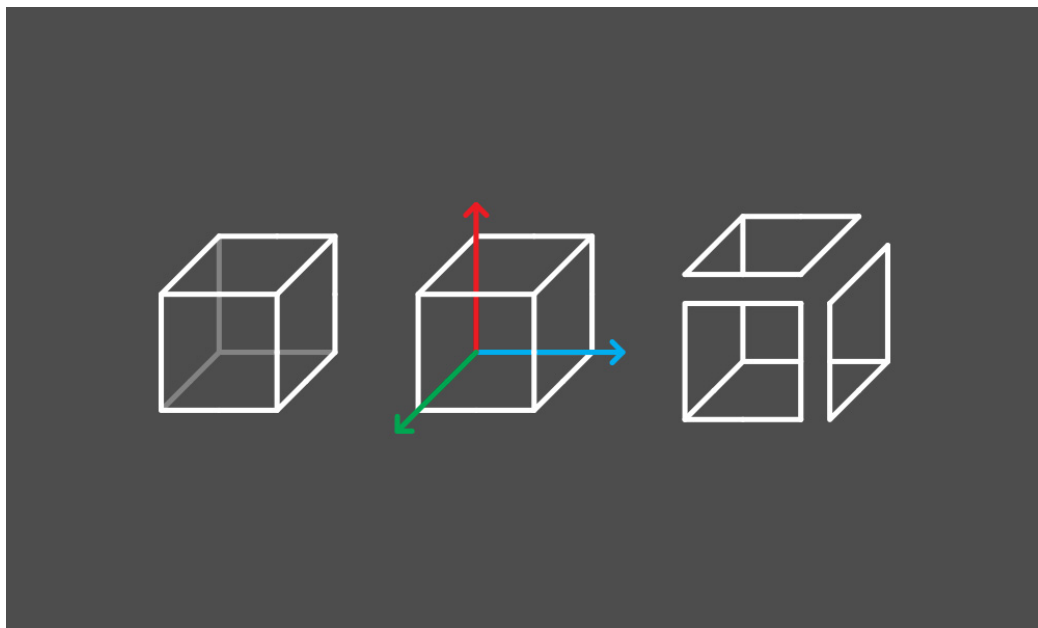
Ena izmed pomanjkljivosti, ki smo jo opazili tekom izdelave spletne strani, je, da tako razstava kot tudi avtor nimata nobenega prepoznavnega elementa ali znaka. Znak je eden izmed glavnih vizualnih komponent celostne grafične podobe kateregakoli podjetja. Znake lahko dan danes najdemo prav na vsakemu mediju od poslovnih vizitk do velikih plakatov, od reklam do spletnih strani. Prav zaradi tega lahko znak z dobrim dizajnom prinese spletni strani veliko uspeha, medtem ko lahko slab znak odvrne potencialne obiskovalce. Znak se mora dobro ujemati s celotno vizualno prezentacijo strani. Najbolje izgleda, kadar je obdan s skladnimi grafičnimi elementi in primerno tipogra-

fijo [12].

Znak služi za prepoznavno v trenutku, ko uporabnik obišče spletno stran. Raziskave kažejo, da je pogled uporabnikov ob pristopu na stran in med brskanjem po vsebinah bolj usmerjen v levo, v kolikor se njihov jezik piše z leve proti desni strani. Zato večina oblikovalcev postavi znak v levi zgornji kot spletne strani, česar smo se poslužili tudi mi pri oblikovanju [16].

Z znakom smo želeli poudariti dva pojma, ki ju lahko najdemo v naslovu razstave: prostor in vržen iz tira (zamaknjen). Prostor smo predstavili z žičnim okvirjem kocke v izometrični perspektivi. Narisani objekt smo obravnavali kot 2D sliko in vsako ploskev kocke zamaknili navzven po določeni dimenziji. S tem smo znaku dodali še motiv zamaknjenosti ter dosegli zastavljeni cilj. Ker je moral biti znak v skladu z dizajnom strani, smo vse črte obarvali s črno oz. belo barvo, odvisno od podlage, na kateri je bil prikazan. Narisali smo ga v programu za vektorsko grafiko in ga izvozili kot SVG. Slednji format smo izbrali zato, ker so SVG slike odzivne že same po sebi in jih je lažje modificirati z JavaScriptom in CSSom.

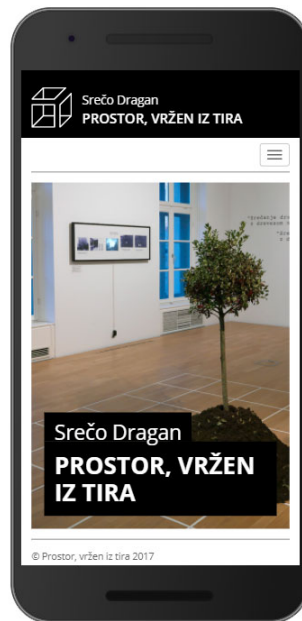
Znak smo nato prenesli na spletno stran ter desno od njega dodali še napis z imenom in priimkom avtorja v zgornji vrstici ter naslovom razstave v spodnji vrstici. Vse skupaj smo nato ovili v a HTML element, ki deluje kot povezava, ki preusmeri uporabnika na vstopno stran. Znak smo uporabili tudi pri izdelavi favicone. To ikono smo naredili kot PNG sliko, pri čemer je znak črne barve, ozadje pa transparentno. Favicon lahko običajno najdemo na levi strani zavihka v brskalniku, poleg tega pa jo nekateri brskalniki prikazujejo tudi, ko uporabnik ustvari bližnjico s povezavo na spletno stran, tako na namizju kot tudi na mobilnih napravah. Res je, da gre za zelo majhno slikico, vendar je doslednost pri podrobnostih tisto, kar naredi spletno stran unikatno.



Slika 4.1: Dizajn znaka za spletno stran.



Slika 4.2: Naslovna stran spletne strani. Prikaz postavitve znaka.



Slika 4.3: Naslovna stran spletne strani na manjšem zaslonu.

4.4 Navigacija

Pomembna komponenta na vsaki spletni strani je tudi navigacija. Ima zelo velik vpliv na velikost prometa na spletni strani, optimizacijske iskalnike in uporabniško izkušnjo. Obiskovalci tipično pričakujejo, da bodo na spletni strani našli navigacijo v horizontalni obliki na vrhu strani ali pa v vertikalni obliki na levem robu. Z implementacijo enega izmed dveh omenjenih načinov navigacije uporabniku olajšamo uporabo strani, saj se navigacija nahaja na že znanemu mestu v že znani obliki. Pomembno je tudi število povezav, ki jih nastavimo v navigaciji. Strani z dolgimi navigacijami delujejo zelo slabo, saj si v kratkoročnem spominu človek uspe zapomniti le do sedem stvari. S krajšimi navigacijami izničimo možnost, da bi obiskovalec spregledal pomembne povezave. Manj kot imamo povezav v navigaciji, bolj bodo vidne. Ne smemo pozabiti tudi na vrstni red. Najpomembnejše povezave pridejo na prva in zadnja mesta, medtem ko manj pomembne postavimo v sredino vrste [27]. V našem primeru smo se odločili, da v glavni meni na vrhu strani nastavimo le povezave do glavnih treh podstrani. Postavili smo jih v naslednjem vrstnem redu: Sobe in projekti, O razstavi in Vizualizacije. Ker bo večina obiskovalcev našo stran obiskala z namenom, da si ogleda dela, ki so bila razstavljeni na razstavi, smo povezavo do te vsebine postavili na prvo mesto. Ker znak deluje tudi kot povezava na vhodno stran, v meni nismo dodali povezave, ki bi storila isto.

Pomembna komponenta, ki je tipa sekundarne navigacijske sheme, so tako imenovane drobtinice. Na spletnih straneh z velikim številom nivojev podstrani lahko drobtinice znatno izboljšajo uporabnikovo prehajanje med njimi. V smislu uporabnosti, drobtinice bistveno zmanjšajo številko klikov, ki jih mora opraviti obiskovalec, da pride do podstrani na višjem nivoju ter naredijo podstrani in sekcije bolj vidne. Nudijo tudi efektivno vizualno pomoč, saj identificirajo, kje se uporabnik trenutno nahaja znotraj hierarhije spletne strani [2]. Zaradi vseh teh prednosti smo drobtinice implementirali na vseh podstraneh sekcije „Sobe in projekti“. Prva drobtinica z nizom »Razstava« preusmeri uporabnika na stran, kjer si lahko izbere sobo iz razstave.

Druga drobtinica z imenom izbrane sobe uporabnika preusmeri na stran, kjer so predstavljeni vsi projekti v tej sobi. Zadnja drobtinica z nizom izbranega projekta pa služi le kot informacija o trenutnem nahajališču uporabnika.

4.5 Gradnja podatkovne baze

Za gradnjo podatkovne baze smo uporabili NoSQL DBMS, MonogDB. MonogDB velja za „najhitrejši rastoči ekosistem podatkovnih baz [21].“ Izbrali smo ga, ker omogoča „hitro gradnjo aplikacij, nadzorovanje veliko različnih tipov podatkov in učinkovito upravljanje aplikacij večjega merila.“ Poleg tega je ena izmed njegovih ključnih prednosti ta, da omogoča fleksibilne podatkovne modele in hitre spremembe v shemi [20].

Za lažjo interakcijo z njim pa smo uporabili knjižnico Mongoose. MonogDB shranjuje podatke kot dokumente, ki so podobni JSON objektom. Vsak dokument ima lahko več različnih polj z vrednostmi. Dokumenti so shranjeni v kolekciji, ki je podobna strukturi tabele pri relacijskih podatkovnih bazah. Z uporabo Mongoosa smo najprej izdelali shemo. „Vsaka shema je povezana z MonogDB kolekcijo in definira obliko dokumentov znotraj nje. Shema vsebuje ključne, ki definirajo lastnosti v našem dokumentu s pripadajočimi shematskimi tipi [22].“ Odločili smo se, da bomo naredili dve shemi. Eno za sobe razstave in eno za projekte. Za shemo sobe smo definirali ključne: številka sobe, število projektov in opis, za shemo projekta pa smo kot ključne definirali vse različne tipe informacij, ki smo jih našli pri pripravi gradiva.

```
var roomSchema = new mongoose.Schema({
  room_number: Number,
  number_of_projects: Number,
  description: String
});

var projectSchema = new mongoose.Schema({
```

```
    authors: [String],
    title: String,
    manufacturing_period: String,
    media: String,
    media_duration: String,
    reconceptualization_degree: String,
    scene: String,
    owner: String,
    description: String,
    room_number: Number,
  });
```

Da lahko uporabimo definirane sheme, jih je potrebno pretvoriti v modele. To storimo s klicem Mongoosove funkcije, ki kot parametra vzame ime modela in shemo. Za naši dve shemi smo uporabili naslednja dva klica:

```
var Room = mongoose.model('Room', roomSchema);
var Project = mongoose.model('Project', projectSchema);
```

Ko smo imeli definirana modela, smo lahko preprosto v kolekcije vnašali instance teh modelov kot dokumente. Primer vnosa enega izmed projektov:

```
Project.create({
  authors: ["Srečo Dragan"],
  title: "Konceptna tabla VII",
  manufacturing_period: "2010",
  media: "Konceptna tabla",
  reconceptualization_degree: "3. rekonceptualizacija",
  description: "Gibljive slike, ki se ...",
  room_number: 1 },
function(err, project){
  if(err) console.log(err);
  else console.log(project)
});
```

4.6 Interaktivni tloris

Pri analizi primerov dobre prakse pri oblikovanju spletnih strani razstav smo ugotovili, da nekatere spletne strani uporabljajo že zelo napredne in vizualno bogate komponente. Tudi sami smo želeli narediti nekaj podobnega za našo spletno stran. Iskali smo interaktivno komponento, ki bi izboljšala uporabniško izkušnjo, kot tudi pustila vtis na obiskovalcih. Vedeli smo, da bo oseba prišla na našo spletno stran zaradi enega izmed naslednjih dveh razlogov: ker razstave še ni videla in bi si rada ogledala Draganova dela ali pa si je razstavo že ogledala in bi si rada osvežila spomin na določene sobe ali projekte. Ko si je oseba prišla ogledat razstavo, je bila na vhodu prisiljena izbrati eno izmed dveh sob, s katero bo začela svoj ogled. Če je izbrala sobo na levi, si je najprej ogledala Draganova najnovejša dela, če je izbrala sobo na desni pa njegova najstarejša dela. V kolikor si je potem želela ogledati še projekte v kakšni drugi sobi, je do njih lahko prišla le s sprehodom iz ene sobe v drugo, dokler ni prispela do željene destinacije. Ker obiskovalce naše strani nismo želeli prisiliti, da morajo na začetku ogleda izbirati le med dvema soba in ker smo hoteli, da je ogled katerekoli sobe oddaljen z enakim številom klikov od katerekoli točke na strani, smo se odločili, da bomo ta problem rešili z izdelavo interaktivnega tlorisa razstave. Ideja je bila naslednja: na strani bo obiskovalec lahko videl tloris nadstropja razstave, pod njim pa slike posamezne sobe. Ko se bo z miško pomaknil na območje sobe, ki si jo želi ogledati, se bo površina te sobe obarvala in njena pripadajoča slika se bo povečala. Slike bomo dodali še napis s številko sobe in številom projektov v njej. Ker predvidevamo, da se bo kateri izmed obiskovalcev najprej z miško pomaknil na slike sob in ne na tloris, bomo animacijo naredili v obe smeri. Torej ne glede na to, ali pride miška najprej v stik s tlorisom ali sliko, se bosta animirala oba para. Ko bo uporabnik kliknil na obarvani tloris sobe ali sliko, se mu bo odprla podstran z vsebino izbrane sobe. Izdelavo smo začeli tako, da smo si najprej iz spletne strani Muzeja sodobne umetnosti Metelkova prenesli PDF datoteko s tlorisom nadstropja, v katerem je bila postavljena razstava. Tloris nadstropja smo spremenili v vektorsko grafiko ter vanjo narisali poligone, pri

čemer je vsak poligon predstavljal površino ene izmed sob. Grafiko smo nato spremenili v SVG. Kot smo že omenili, je ena izmed prednosti slik v tem formatu, da so odzivne, poleg tega pa jih lahko tudi zapišemo kot HTML kodo ter jo zelo preprosto dodamo preostali kodi strani. Potem je bilo potrebno HTML elementom dodati nek atribut, preko katerega smo lahko ugotovili, nad katerim parom se izvaja animacija. Odločili smo se uporabiti data atribut, ki dopušča hranjenje podatkov po meri. Data atribut smo poimenovali „data-room-number“ ter mu kot vrednost podali niz s številko sobe. Nato smo atribut dodali vsem elementom poligonov, ki so predstavljali površine sob in vsem div elementom z img elementi, ki so vsebovali njihove slike.

Primer enega izmed parov:

```
<a href="/soba/2" target="_self">
  <polygon class="room"
    data-room-number="2"
    fill="transparent"
    points="553.4,162.3 549.4,162.3 549.4,158.3 ...">
  </polygon>
</a>

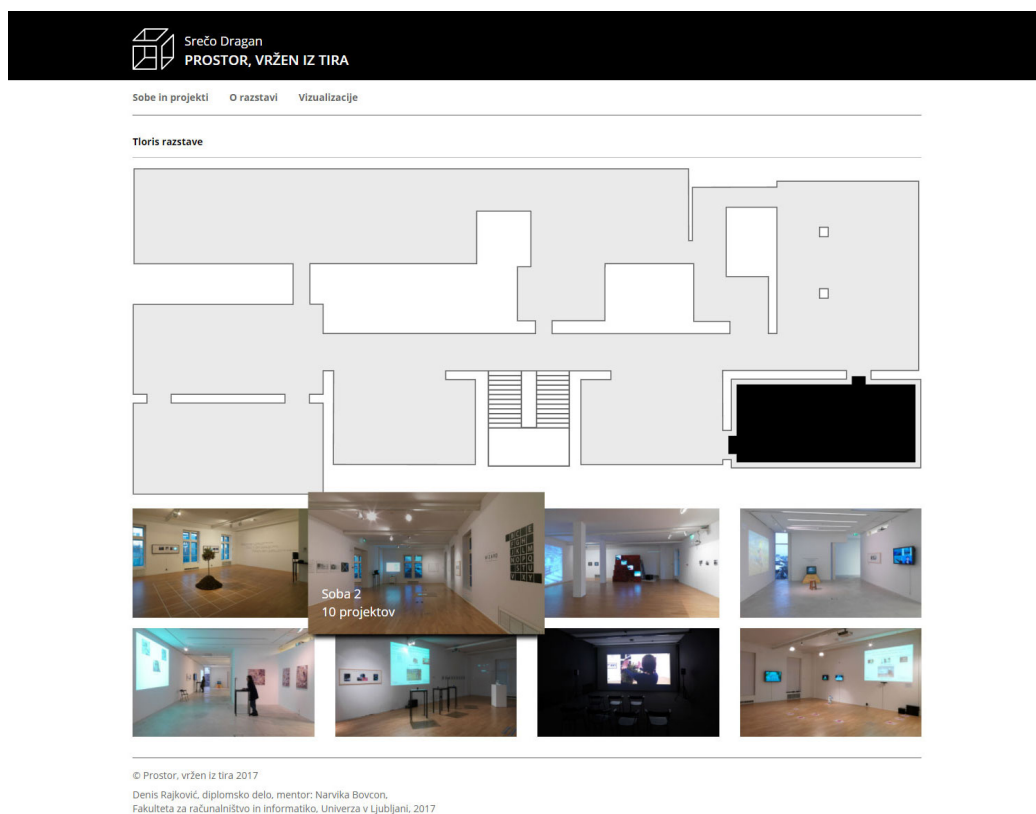
<a href="/soba/2">
  <div class="img-container" data-room-number="2">
    
    <div class="overlay">
      <div class="text">
        Soba 2 <br>10 projektov
      </div>
    </div>
  </div>
</a>
```

Nato smo na elemente nastavili poslušalce za dogodke, ko na njihovo območje vstopi in izstopi miška. Elemente smo poiskali preko njihovih CSS razredov z jQuery selektorji. V kolikor je šlo za dogodek vstopa, smo elemente animirali z dodajanjem novih CSS razredov, pri izstopu pa smo jih odstranjevali. Pri tem smo uporabili kar nekaj modulov, ki so prišli z izdajo nove verzije CSSa. Elementom smo dodajali sence, jih skalirali in jih animirali z gladkimi prehodi. Končni rezultat so bile vizualno zanimive animacije.

```
$('.img-container')
.on('mouseover', function () {
    var roomNumber = $(this).data("room-number");
    var roomPolygon = $('.room[data-room-number="'
    + roomNumber + '"]');
    roomPolygon.addClass('room-hover');})
.on('mouseleave', function () {
    var roomNumber = $(this).data("room-number");
    var roomPolygon = $('.room.room-hover[data-room-number="'
    + roomNumber + '"]');
    roomPolygon.removeClass('room-hover');
});

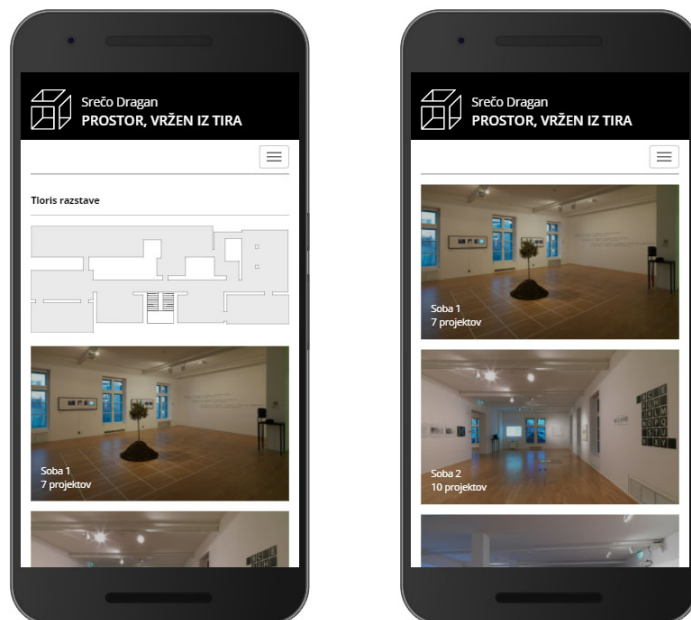
$('.room')
.on('mouseover', function () {
    var roomNumber = $(this).data("room-number");
    var roomImgContainer =
    $('.img-container[data-room-number="' + roomNumber + '"]');
    roomImgContainer.addClass('img-container-hover');})
.on('mouseleave', function () {
    var roomNumber = $(this).data("room-number");
    var roomImgContainer =
    $('.img-container.img-container-hover[data-room-number="'
    + roomNumber + '"]');
```

```
roomImgContainer.removeClass('img-container-hover');
});
```



Slika 4.4: Interaktivni tloris na pogledu Sobe in projekti. Prikaz animacije.

Ko smo testirali animacije še na manjših zaslonih, smo opazili, da pri mobilnih napravah zaslon ni dovolj velik, da bi nanj na pregleden način postavili vse elemente. Ker animacija izgubi svoj pomen, v kolikor uporabnik ne more istočasno videti tako tlorisa kot tudi slik, smo se odločili, da na manjših zaslonih odstranimo eno izmed komponent. Ker so območja sob na manjših zaslonih tako majhna, da jih je težko klikniti, smo se odločili, da raje obdržimo slike sob. Slednje smo nato raztegnili čez celotno širino zaslona in jih zložili eno pod drugo.




Slika 4.5: Prikaz komponent na manjšem zaslonu.

4.7 Pogledi

4.7.1 Soba


Ko smo dokončali podstran s tlorisom razstave, je bilo pomembno narediti še poglede, ki jih uporabnik vidi, ko izbere eno izmed sob. Na pogledu posamezne sobe smo na vrhu najprej namestili že prej omenjeno komponento drobtinice. S pomočjo te komponente lahko uporabnik razbere, katero sobo si trenutno ogleduje. Pod njo smo dodali še naslov s številko sobe in podnaslov, v katerem smo navedli število projektov v tej sobi. Ker v tem pogledu nismo želeli le naštetih projektov, ki so bili razstavljeni v sobi, smo strani dodali še sliko sobe in kratek opis. S pomočjo slike lahko uporabnik vidi, kako je izgledala celotna postavitve projektov v sobi in s tem dobi boljši občutek, kaj mu je želel umetnik sporočiti v njej. Projekte v sobi smo želeli prikazati na tak način, da bi uporabnik že na tej točki dobil delni vpogled v njihovo vsebino in kako so izgledali. Istočasno smo vedeli, da za posamezni projekt ne smemo navesti preveč informacij, kajti v temu pogledu je bilo ključno to, da lahko uporabnik naredi hiter pregled nad vsemi projekti in se na podlagi tega odloči, katerega si bo ogledal. Tako smo se odločili, da za vsak projekt nastavimo sliko, ime projekta in le prvih nekaj besed iz njegovega opisa. V kolikor uporabnik klikne na sliko ali ime projekta, pa ga stran preusmeri na pogled s celotno vsebino izbranega projekta.


Srečo Dragan
PROSTOR, VRŽEN IZ TIRA

[Sobe in projekti](#)
[O razstavi](#)
[Vizualizacije](#)


Razstava / Soba 1

Soba 1
 7 projektov




Takrat je bilo to za nas zelo pomembno in smo rekli: programator Srečo Dragan, programatorka Nuša Dragan, gre za premik konec šestdesetih let, za programirano umetnost


Projekti




Belo mleko belih prsi
 Statičen črno-bel posnetek »slušno govorne grupne komunikacije zlogov«, ki je obveljal za prvi video...




72 h zime in 72 h jeseni
 Srečanje drug nasproti drugemu brez dotika z drevesom na sredi v parku Tivoli. Srečanje drug nasprot...




72 h zime in 72 h jeseni
 Srečanje drug nasproti drugemu brez dotika z drevesom na sredi v parku Tivoli. Srečanje drug nasprot...



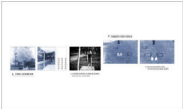
72 h zime in 72 h jeseni
 Rekonstrukcija performansa 72 h zime 72 h jeseni in istosimenskega filma avtorjev Nuše & Sreča Dragan...



OHO, Atelje '69
 Na razstavi v Moderni galeriji so ohojevci stopnjevali novo estetiko »samovoljne« poietične selekcije...



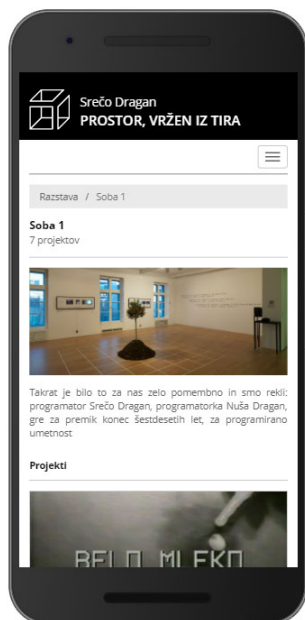
Konceptna tabla VI
 Konceptualna umetnost, artikulirana v medijih teksta, fotografije, filma, happeninga in odtisa v rev...



Konceptna tabla VII
 Gibljive slike, ki se kontekstualizirajo kot perceptivne in programirane situacije performansa v med...

© Prostor, vržen iz tira 2017
 Denis Rajkovič, diplomsko delo, mentor: Narvika Bovcon,
 Fakulteta za računalništvo in informatiko, Univerza v Ljubljani, 2017

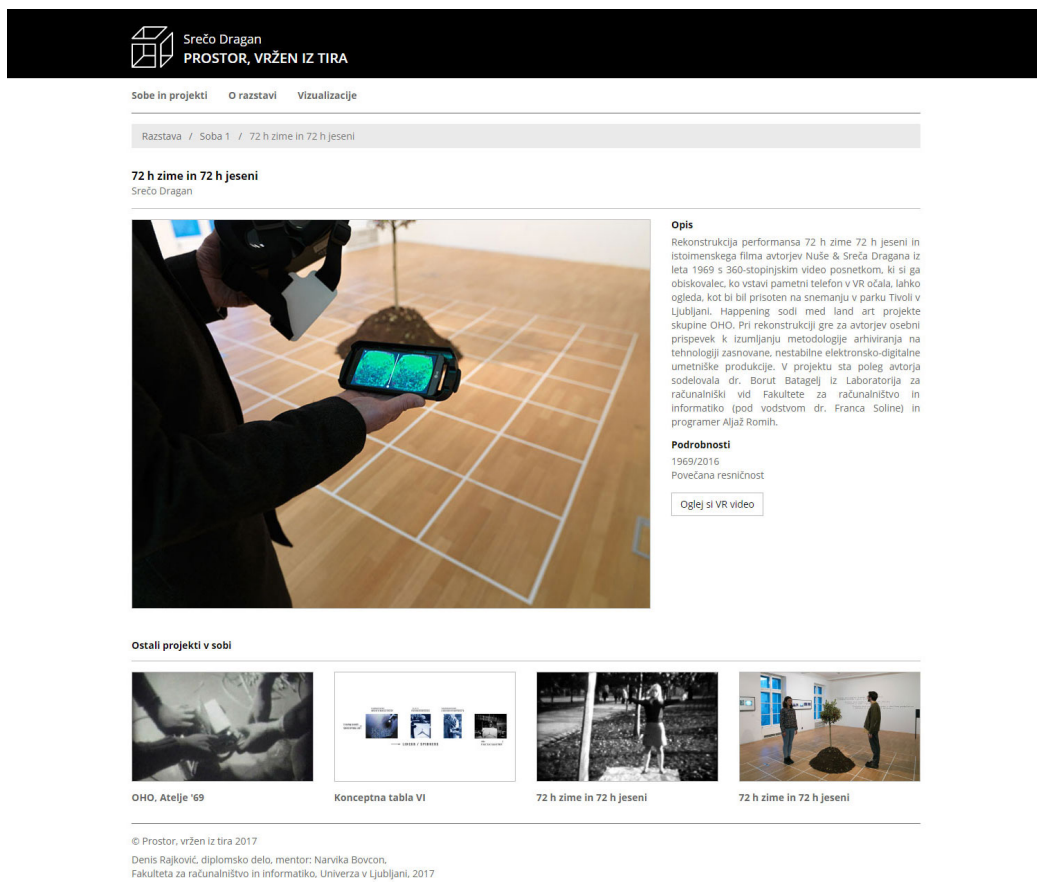
Slika 4.6: Pogled ene izmed sob.



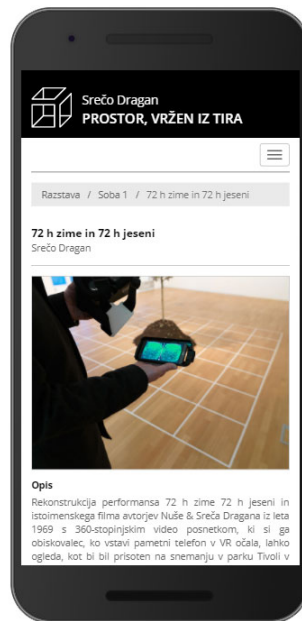
Slika 4.7: Pogled ene izmed sob na manjšem zaslonu.

4.7.2 Projekt

Tudi v pogledu posameznega projekta smo na vrhu dodali komponento drobtinice pri čemer smo kot zadnjo povezavo navedli ime projekta. Pod njo je sledil naslov z imenom projekta, v podnaslovu pa smo našli njegove avtorje. Slednji informaciji smo smatrali kot najbolj pomembni, zato smo ju navedli najprej in jima dodelili celotno vrstico zase. Nato smo prostor za vsebino razdelili v razmerju 2:1, ter večji del dodelili sliki projekta. Razlog za to je bil, da si bo večina obiskovalcev na stran raje prišla ogledat slike kot prebrat vsebino, zato je smiselno, da na pogledu dobijo več prostora. Desno od slike smo navedli še informacije o projektu. Razdelili smo jih v dve sekciji: opis in podrobnosti. V sekcijo opis smo dodali opis projekta, pod podrobnosti pa vse ostale informacije o projektu, ki smo jih hranili v bazi. V kolikor je projekt vseboval tudi video, smo pod informacijami na desni strani dodali še gumb z nizom „Oglej si video“. Ob kliku na gumb smo sprogramirali, da se na ekranu pojavi modalno okno z videom, ki ga lahko uporabnik poveča na velikost celotnega ekrana. Ker se nam je zdela povečava medija čez celotni zaslon uporabna funkcionalnost, smo jo dodali tudi sliki. Sliko lahko uporabnik poveča s preprostim klikom nanjo. Če smo za določen projekt v bazi hranili več slik, smo te v stolpcih izrisali v naslednji vrstici. Tudi tem slikam smo dodali funkcionalnost povečave. Na tej točki smo imeli občutek, da strani manjka še nekaj, kar bi uporabnika nagovarjalo, da si ogleda še kakšen projekt iz iste sobe. Odločili smo se, da pod predstavitev projekta dodamo še sekcijo z ostalimi projekti v sobi. V tej sekciji smo naključno izbrali štiri projekte, ki so se nahajali v isti sobi, ter jih kot na pogledu sobe, razvrstili v stolpce s sliko in imenom projekta. V kolikor si uporabnik želi ogledati enega izmed njih, to stori s preprostim klikom na sliko ali ime.



Slika 4.8: Pogled enega izmed projektov.



Slika 4.9: Pogled enega izmed projektov na manjšem zaslonu.

Poglavje 5

Optimizacija in izboljšave

Zaradi nenehnega razvoja in nadgradenj aplikacij le te sčasoma rastejo po obsegu in funkcionalnostih, vendar pa se lahko zgodi, da količina podatkov, ki jo prenaša aplikacija, povzroči njeno počasno delovanje. Zato je že pri prvem oblikovanju potrebno poskrbeti za njeno optimizacijo ter temu cilju slediti tudi pri vseh nadgradnjah [31].

5.1 Slike

Večino prenesenih bajtov na spletni strani predstavljajo slike, ki pogosto zasedajo veliko vizualnega prostora, zato lahko njihova optimizacija prinese največ prihranka in s tem izboljša zmogljivost in odzivnost spletne strani. Pri iskanju optimalnih nastavitev za vsako sliko pa je potrebno upoštevati več dimenzij kot so: vsebina kodiranih podatkov, kakovost, zmogljivosti formata, dimenzije pikslov in drugo, poleg tega pa se je potrebno zavedati, da je pri optimizaciji slik potrebno slediti tako znanstvenemu kot umetniškemu vidiku; umetniški vidik, saj ni nobenega dokončnega odgovora za najboljšo kompresijo posamezne slike, in znanstveni, saj obstaja veliko dobrih tehnik in algoritmov, ki lahko bistveno zmanjšajo velikost slike [11]. Povod za optimizacijo slikovnega gradiva je bila ugotovitev, da se pogledi s slikami projektov izrisujejo zelo počasi ravno zaradi časa, ki ga potrebuje brskalnik,

da jih prenese in izriše.

Če smo hoteli odpraviti ta problem, smo morali najprej narediti analizo strani in preračunati, kakšne dimenzije slik sploh potrebujemo. Tega smo se lotili tako, da smo za vsak pogled s projekti preverili, kakšna je največja dimenzija slike, ki jo prikažemo. Pri tej analizi nismo pozabili, da se zaradi odzivnega dizajna spreminja tudi dimenzija slike glede na velikost zaslona na istem pogledu. Prišli smo do zaključka, da bo potrebno za vsako sliko narediti tri slike različnih dimenzij. Ker smo želeli slike tudi optimizirati, smo za to opravilo uporabili program Photoshop. V Photoshopu imamo poleg navadnega shranjevanja tudi opcijo, da sliko shranimo za splet. Uporaba slednje opcije je zelo pomembna, saj je lahko končna slika do trikrat manjša v velikosti prostora, ki ga zasede, v primerjavi z navadnim shranjevanjem. Ko v programu izberemo to opcijo, se nam odpre okno, v katerem poleg dimenzij določimo še tip slike in kvaliteto slike. Za tip smo vedno izbrali JPEG, saj je znan po tem, da uporablja kombinacijo različnih optimizacij, s čimer zmanjša velikost bajtov datoteke. Pri testiranju shranjevanja z različnimi nastavitvami kvalitete smo ugotovili, da se pri nastavitvi visoke kvalitete skoraj da ne opazi kakršnekoli degradacije v pikslih, zato smo to nastavitvev uporabili v večini primerov.

Končni rezultat so bile tri skupine slik z naslednjimi lastnostmi:

- Prvo skupino slik smo poimenovali „thumbnail“ slike. To so bile slike dimenzij 750x450 pikslov. Te slike smo uporabili na pogledu sobe za naštevanje projektov in na pogledu projekta v sekciji „ostali projekti“.
- Drugo skupino slik smo poimenovali „detail“ slike. To so bile slike, pri katerih smo zahtevali, da je njihova širina 750 pikslov, višino pa smo prilagodili glede na razmerje dimenzij originalne slike. Te slike smo uporabili na pogledu projekta kot predstavitvene slike.

- Zadnjo skupino slik pa smo poimenovali „full“ slike. To so bile slike, ki so se prikazale, če je uporabnik kliknil nanjo v pogledu projekta in smo jo izrisali čez celotni zaslon. Glede na to, da se velikosti zaslonov elektronskih naprav dan danes spreminjajo zelo hitro, je nemogoče oceniti njihovo maksimalno širino. Zaradi tega smo se odločili za neko fiksno vrednost in slike naredili s širino največ 1200 pikslov. Višino smo prilagodili glede na razmerje dimenzij originalne slike.

V HTMLjih smo nato za vsak `img` element definirali vir slike iz ustrezne skupine ter s tem dosegli, da brskalnik na vsakem pogledu naredi poizvedbo samo po slikah primernih velikosti in tako prenaša le potrebno število bajtov. Pogledi so se zato začeli izrisovati bistveno hitreje. Kljub temu, da smo iz vsake slike naredili tri nove, se je skupno število bajtov vseh slik, ki jih uporabljamo na spletni strani zmanjšalo skoraj za 99%. Natančneje, na začetku je skupno število bajtov znašalo približno 2,95 GB, po optimizaciji pa samo nekaj čez 30 MB. S tem smo tudi bistveno zmanjšali zahtevo po prostoru na strežniku.

5.2 Videi

Ker smo z optimizacijo slik dosegli zelo veliko izboljšavo spletne strani, smo se odločili, da poskusimo optimizirati tudi video gradivo. Videe smo prejeli v več različnih formatih, kot so AVI, MPG, MP4 in TS.

V preteklih letih so na spletu prevladovali medijski predvajalniki na osnovi Flasha s podporo FLV video formatov, danes pa je najbolj priljubljen HTML5 video element. Najprej smo raziskali, katere video formate podpira slednji in ugotovili, da so to formati MP4, WebM in Ogg. Idealno bi bilo, da bi za vsak video imeli vse tri formate, toda ugotovili smo, da to ni potrebno, ker lahko vse brskalnike podpremo z uporabo le dveh [9].

Brskalniki podpirajo naslednje formate [9]:

- Mozilla Firefox – WebM, Ogg
- Google Chrome – WebM, Ogg
- Opera – WebM, Ogg
- Safari – MP4
- Internet Explorer 9 – MP4
- Internet Explorer 6 – 8 – ne podpira HTML5, samo Flash

Ker je WebM sponzoriran s strani Googla in ker je trenutno zelo priljubljen v HTML5 skupnosti, smo se za podporo prvih treh brskalnikov na seznamu odločili, da uporabimo tega. „WebM je format odprtih medijskih datotek, ustvarjen za splet. Datoteke WebM sestavljajo video prenosi, kompresirani z video kodekom VP8 ali VP9, zvočni prenosi, kompresirani z avdio kodeki Vorbis ali Opus, in teksti formata WebVTT. Ključni razlog, da je ta format tako priljubljen, je v tem, da so njegove temeljne tehnologije (HTML, HTTP in TCP/IP) prosto dostopne in brezplačne za implementacijo.“ WebM je priljubljen tudi zaradi tega, ker kodeka VP8 in VP9 ponujata videe visoke kvalitete z majhno zahtevo po prostoru [38].

Če smo želeli podpreti videe za uporabo še na zadnjih treh brskalnikih na seznamu, pa smo morali uporabiti MP4 video format. MP4 je digitalni multimedijski vsebnik, ki vsebuje video format MPEG-4 z video kodekom H.264 in avdio format MP3 s kodekom ACC. MPEG-4 video format z video kodekom H.264 deluje tudi kot alternativni vir za Flash predvajalnike. To pomeni, da nam za podporo teh predvajalnikov ni potrebno hraniti kopije videov v FLV formatu. MP4 format je podprt tudi s strani iOS in Android naprav, zato je uporaba tega formata zelo pomembna, če želimo, da si uporabniki lahko našo stran ogledajo tudi na teh napravah [23][9].

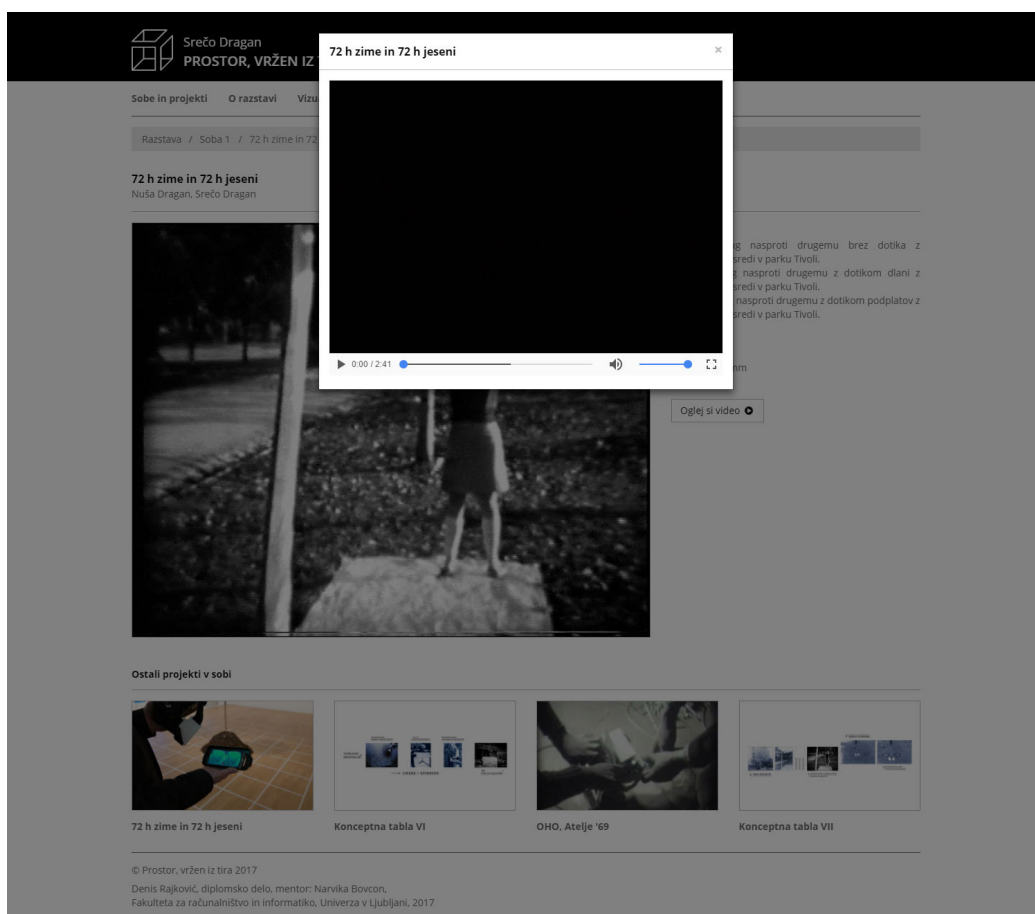
Videe smo zakodirali s pomočjo programa FFMpeg. Za WebM format smo uporabili video kodek VP8 in avdio kodek Vorbis, za MP4 format pa video kodek H.264 in ACC avdio kodek.

Podobno kot pri slikah se je tudi tukaj precej zmanjšalo število bajtov. Videi so na začetku skupaj zavzemali 2,57 GB prostora, po konverziji v druga

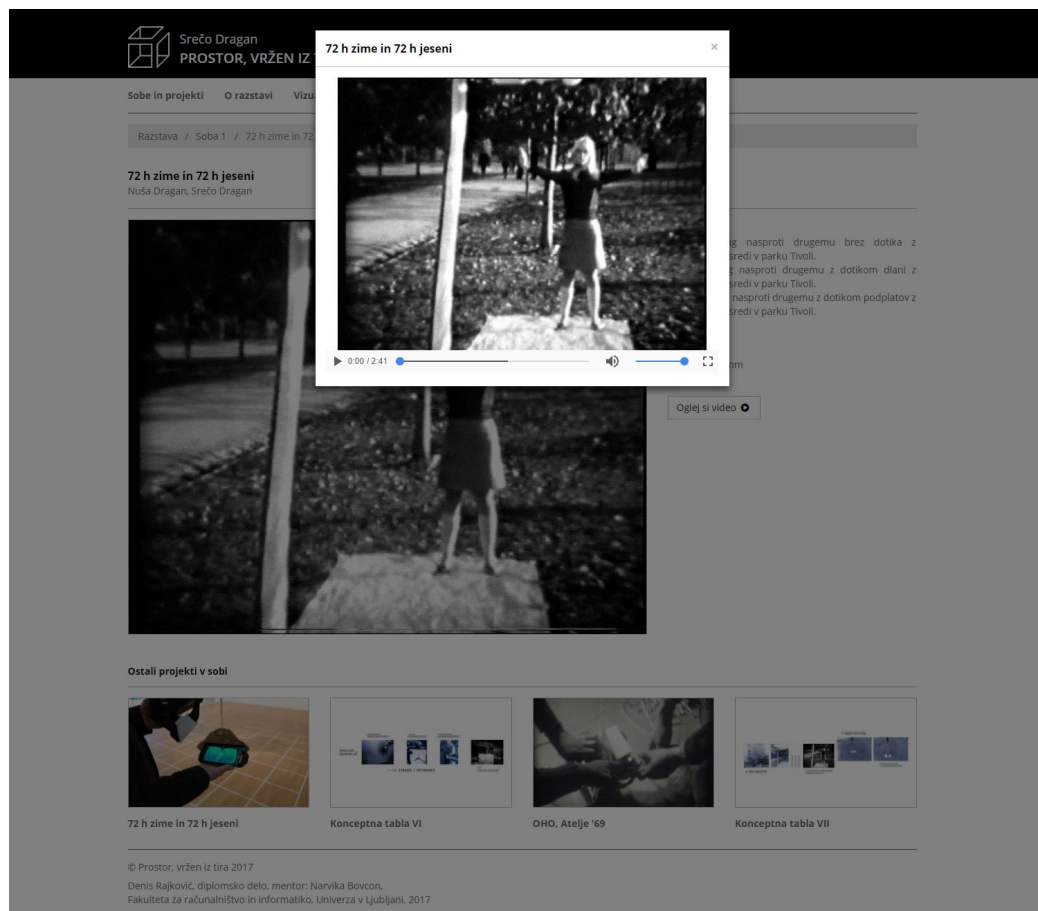
dva formata pa le 1,24 GB, kar je približno 52% manj prostora.

Ker vsi brskalniki ne podpirajo istih video formatov, smo v HTMLju uporabili source element, s katerim lahko za vir videa „določimo več različnih formatov v primerih, ko brskalniki ne podpira enega izmed njih.“ „Dopuščanje brskalniku, da izbere format, je bolj preprosto, hitrejše in potencialno bolj zanesljivo.“ Source elementu smo dodali še type atribut, s katerim pomagamo brskalniku pri odločitvi, kateri tip datoteke prenesti in jo predvajati. Z določitvijo tega atributa za „vsak vir izboljšamo tudi omrežno učinkovitost; brskalniki lahko izbere video vir, ne da bi najprej prenesel košček videa in zaznal njegov format.“ Video elementu smo dodali še atribut poster, s katerim določimo, katera slika se prikaže v oknu videa. S to sliko lahko uporabniku pokažemo vsebino videa še preden ga je brskalniki prenesel ali ga začel predvajati. Poster lahko deluje tudi kot alternativa v primerih, ko vir ni delujoč ali ko brskalniki ne podpira nobenega izmed formatov videa, navedenih med viri [37].

```
<video controlslist="nodownload"
  poster="/assets/images/projects/72-h-zime-in-72-h-
jeseni-2-poster.jpg">
  <source class="webm"
    src="/assets/videos/72-h-zime-in-72-h-jeseni-2.webm"
    type="video/webm">
  <source class="mp4"
    src="/assets/videos/72-h-zime-in-72-h-jeseni-2.mp4"
    type="video/mp4">
  <p>This browser does not support the video element.</p>
</video>
```



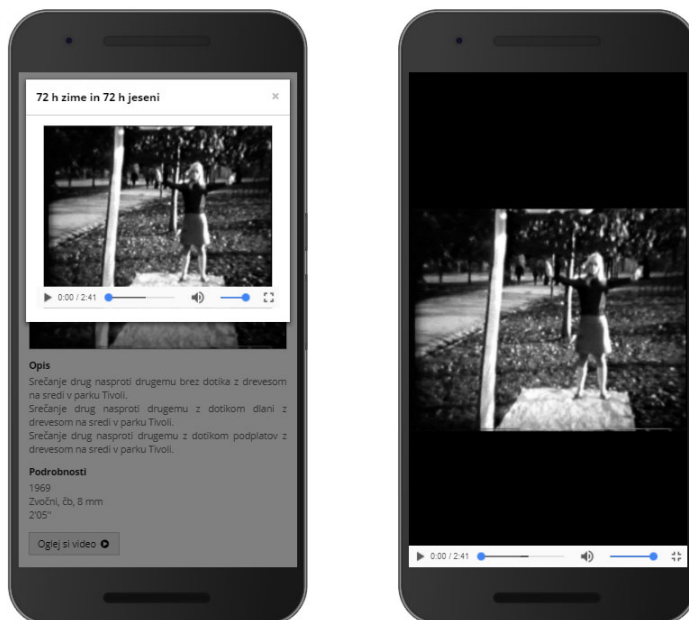
Slika 5.1: Prikaz videa brez poster slike.



Slika 5.2: Prikaz videa s poster sliko.



Slika 5.3: Prikaz videa na celotnem zaslonu.



Slika 5.4: Prikaz videa na manjšem zaslonu.

Poglavje 6

Vzdrževanje in predlogi za prihodnost

Danes si je težko predstavljati področje življenja, v katerega ne posega internet, zato se je njegova narava zelo spremenila in z njegovim še vedno strmim razvojem sklepamo, da se bo ta trend nadaljeval [36]. Zaradi tega je težko napovedati, katere tehnologije, naprave in dizajni bodo aktualni v bližnji in daljni prihodnosti.

Vsekakor je priporočljivo, da se ob izdaji novih verzij tehnologij slednje zamenja s trenutnimi verzijami ter se s tem posodobi tehnologije, ki smo jih uporabili na spletni strani. Tako zagotovimo, da bo spletna stran uporabljala njihove najnovejše nadgradnje, ki se trudijo izboljšati zmogljivost, uporabniško izkušnjo in mnogo drugih področij spletnih aplikacij. Ker je bila ena izmed tehnologij, ki smo jih uporabili na strani, Node.js, lahko za nadgradnje izkoristimo funkcionalnost upravitelja paketov NPM, ki je del Noda. To storimo s preprostim ukazom `npm update`, ki bo posodobil vse pakete, ki jih uporabljamo v aplikaciji, kot tudi naložil manjkajoče. Kot primer opisa postopka posodobitve lahko navedemo že prej omenjeno novejšo verzijo ogrodja Bootstrap. Slednja verzija je trenutno že prosto dostopna, vendar je še v beta verziji in je zato nismo uporabili v našem projektu. Glede na priljubljenost in število aplikacij, ki uporabljajo starejšo verzijo ogrodja, lahko sklepamo,

da bo tudi verzija Bootstrap 4 deležna takega uspeha. Ob uradni izdaji nove verzije predlagamo, da se najprej pregleda njeno dokumentacijo in poišče vse novosti, ki bi bile koristne za našo spletno stran. Sledi posodobitev ogrodja na novejšo verzijo z uporabo upravitelja paketov NPM, na koncu pa moramo, v primeru zaznave napak, napisati še ustrezne CSS in JavaScript popravke v kodi, ki smo jo napisali sami.

S časom poleg razvoja obstoječih na plan prihajajo tudi nove tehnologije. Opazamo, da tehnologija virtualne realnosti postaja vse bolj in bolj popularna [36]. Na razstavi smo videli tudi projekt, pri katerem je bila potrebna uporaba VR očal. „Glavna značilnost te tehnologije je, da uporabnik vidi, sliši in občuti okolje, kot da bi bil v njem resnično prisoten. Bistveno pa je tudi to, da uporabnik ni samo prisoten v tem okolju, ampak je v njem tudi aktiven“, saj je koncept razstave temeljil na ideji, da umetnost nagovarja gledalca k nekemu razmišljanju in gibanju [26]. Ker virtualna tehnologija omogoča gibanje v prostoru, razstava pa nagovarja obiskovalce k temu, vidimo v prihodnosti velike priložnosti za nadgradnjo spletne strani s to tehnologijo. Razstava funkcionira v nekem prostoru in določenem časovnem obdobju (od otvoritve do zaprtja). S spletno stranjo smo razstavo podaljšali na spletu, z dodano implementacijo VR tehnologije pa bi lahko razširili tudi njeno funkcionalnost in zagotovili skoraj identično uporabniško izkušnjo, kot jo je imel nekdo, ki si je šel razstavo ogledat v živo.

Poglavje 7

Zaključek

V okviru diplomske naloge smo izdelali spletno stran za razstavo *Prostor, vržen iz tira* Sreča Dragana. Na spletni strani smo poskušali ustvariti čim bolj podobno izkušnjo, kot jo je lahko imel nekdo, ki si je šel razstavo ogledat v živo. Obiskovalcem ponujamo ogled Draganovih del skozi njegovo petdesetletno zgodovino medijske ustvarjalnosti in njen vpliv na razvoj v slovenskem kulturnem prostoru. Ker je bilo na razstavi možno videti dela na različnih medijih, smo te projekte spremenili v obliko, ki si jo je mogoče ogledati preko spleta. Nismo pozabili tudi na odziven dizajn, ki zagotavlja optimalno izkušnjo, ne glede na to, s katero napravo gledalci dostopajo do spletne strani. S spletno stranjo želimo vzpodbuditi čim več umetnikov, da svoje razstave in dela spremenijo v virtualno obliko ter jih s tem na spletu naredijo dostopne širši publiki in v daljšem časovnem obdobju.

Literatura

- [1] Bootstrap Wikipedia. Dosegljivo: [https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)). [Dostopano 29. 7. 2017].
- [2] Breadcrumbs in web design. Dosegljivo: <https://www.smashingmagazine.com/2009/03/breadcrumbs-in-web-design-examples-and-best-practices/>. [Dostopano 10. 8. 2017].
- [3] CSS Wikipedia. Dosegljivo: https://en.wikipedia.org/wiki/Cascading_Style_Sheets. [Dostopano 29. 7. 2017].
- [4] CSS3. Dosegljivo: <https://developer.mozilla.org/en/docs/Web/CSS/CSS3>. [Dostopano 29. 7. 2017].
- [5] EJS. Dosegljivo: <http://www.embeddedjs.com/>. [Dostopano 10. 8. 2017].
- [6] Essential elements of minimalist website. Dosegljivo: <http://www.creativebloq.com/ux/4-essential-elements-minimalist-website-81516258>. [Dostopano 10. 8. 2017].
- [7] Express. Dosegljivo: https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm. [Dostopano 29. 7. 2017].
- [8] HTML Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/HTML>. [Dostopano 29. 7. 2017].
- [9] HTML5 Videos. Dosegljivo: <http://www.hongkiat.com/blog/html5-videos-things-you-need-to-know/>. [Dostopano 15. 8. 2017].

-
- [10] HTML5 Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/HTML5>. [Dostopano 29. 7. 2017].
 - [11] Image Optimization. Dosegljivo: <https://developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/image-optimization>. [Dostopano 10. 8. 2017].
 - [12] Importance of Logos in Business. Dosegljivo: <http://smallbusiness.chron.com/importance-logos-business-577.html>. [Dostopano 10. 8. 2017].
 - [13] JavaScript Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/JavaScript>. [Dostopano 29. 7. 2017].
 - [14] JavaScriptMVC Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/JavaScriptMVC>. [Dostopano 10. 8. 2017].
 - [15] JQuery Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/JQuery>. [Dostopano 29. 7. 2017].
 - [16] Logo placement. Dosegljivo: <https://www.nngroup.com/articles/logo-placement-brand-recall/>. [Dostopano 10. 8. 2017].
 - [17] Moderna Museet. Dosegljivo: <http://www.modernamuseet.se/stockholm/en/>. [Dostopano 2. 8. 2017].
 - [18] Moderna Museet About. Dosegljivo: <http://www.modernamuseet.se/stockholm/en/about/>. [Dostopano 2. 8. 2017].
 - [19] Moderna Museet Collection. Dosegljivo: <http://sis.modernamuseet.se/about?t:state:flow=2ee4b20f-dc75-4167-bb48-3b98e4484497>. [Dostopano 2. 8. 2017].
 - [20] MongoDB and MySQL Compared. Dosegljivo: <https://www.mongodb.com/compare/mongodb-mysql?jmp=docs>. [Dostopano 29. 7. 2017].

-
- [21] MongoDB System Properties. Dosegljivo: <https://db-engines.com/en/system/MongoDB>. [Dostopano 15. 8. 2017].
- [22] Mongoose. Dosegljivo: <http://mongoosejs.com/docs/guide.html>. [Dostopano 10. 8. 2017].
- [23] MPEG4 vs MP4. Dosegljivo: <https://www.winxdvd.com/answers/mpeg4-vs-mp4.htm>. [Dostopano 15. 8. 2017].
- [24] MVC. Dosegljivo: https://ucilnica1516.fri.uni-lj.si/pluginfile.php/24483/mod_label/intro/12-AJAX.pdf. [Dostopano 10. 8. 2017].
- [25] Narodna galerija Slovenije. Dosegljivo: <http://www.ng-slo.si/si/>. [Dostopano 2. 8. 2017].
- [26] Navidezna resničnost. Dosegljivo: https://sl.wikipedia.org/wiki/Navidezna_resni%C4%8Dnost. [Dostopano 15. 8. 2017].
- [27] Navigation mistakes. Dosegljivo: <https://blog.kissmetrics.com/common-website-navigation-mistakes/>. [Dostopano 10. 8. 2017].
- [28] Node.js. Dosegljivo: https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm. [Dostopano 29. 7. 2017].
- [29] O Narodni galeriji. Dosegljivo: <http://www.ng-slo.si/si/o-narodni-galeriji>. [Dostopano 2. 8. 2017].
- [30] Object Modeling in Node.js with Mongoose. Dosegljivo: <https://devcenter.heroku.com/articles/nodejs-mongoose>. [Dostopano 29. 7. 2017].
- [31] Optimizing Content Efficiency. Dosegljivo: <https://developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/>. [Dostopano 10. 8. 2017].

-
- [32] Responsive design. Dosegljivo: <https://www.optiweb.com/responsive-design/>. [Dostopano 3. 8. 2017].
- [33] Srečo Dragan – Prostor, vržen iz tira. Dosegljivo: <http://www.mglj.si/si/razstave/1905/sreco-dragan-prostor-vrzen-iz-tira/>. [Dostopano 18. 8. 2017].
- [34] Tate. Dosegljivo: <http://www.tate.org.uk/>. [Dostopano 2. 8. 2017].
- [35] Tate Wikipedia. Dosegljivo: <https://en.wikipedia.org/wiki/Tate>. [Dostopano 2. 8. 2017].
- [36] The Future of Design. Dosegljivo: <https://www.awwwards.com/the-future-of-design-according-to-7-web-visionaries.html>. [Dostopano 15. 8. 2017].
- [37] Video. Dosegljivo: <https://developers.google.com/web/fundamentals/media/video>. [Dostopano 10. 8. 2017].
- [38] WebM. Dosegljivo: <https://www.webmproject.org/about/faq/>. [Dostopano 15. 8. 2017].
- [39] Jon Duckett. *HTML and CSS: Design and Build Websites*. Wrox, 2011.